Energy, Mines and Resources Canada
Énergie, Mines et Ressources Canada

CANMET

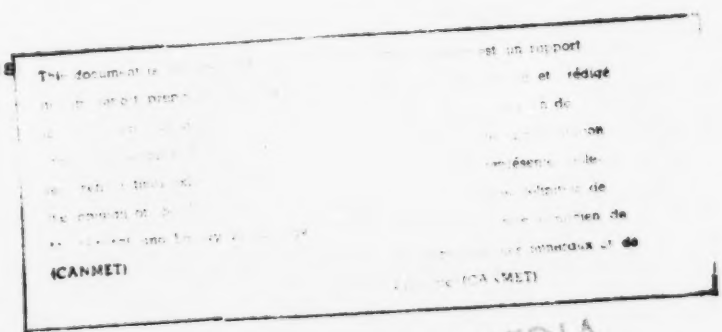
Canada Centre for Mineral and Energy Technology
Centre canadien de la technologie des minéraux et de l'énergie

FIVE CCL PROCEDURES GREATLY SIMPLIFYING THIS LABORATORY'S FILE TRANSFER REQUIREMENTS BETWEEN MICROCOMPUTERS AND THE CYBER

P.L. Bedard and K. Bartels
Metallurgical Chemistry Section
Extractive Metallurgy Laboratory

January 1985

Project MRP-3.4.0.2.02
Hydrometallurgical Lead Process



24 pp

MINERALS RESEARCH PROGRAM
MINERAL SCIENCES LABORATORIES
DIVISION REPORT MRP/MSL 85-8 (TR)

MICROMEDIA
MAR 15 1985

FIVE CCL PROCEDURES GREATLY SIMPLIFYING THIS LABORATORY'S FILE TRANSFER REQUIREMENTS BETWEEN MICROCOMPUTERS AND THE CYBER

by

P.L. Bedard* and K. Bartels**

ABSTRACT

Five CYBER Control Language routines have been written and are described in this report which greatly simplify file transfers between microcomputers and the CYBER, by making it unnecessary for users to type in a series of complicated command strings each time a file transfer is required.

The procedures ULXFNC, ULXFC, UL..., DLXF and DLNXF prompt the user and provide limited help levels, so that users may easily choose the correct and appropriate parameters required by the CYBER communications software XMODEM and the file conversion software XFORMAT. Each procedure is fully described, and detailed installation instructions have been provided, whereby users may easily implement the use of these procedures whenever they log on the to CYBER. In addition, a sixth procedure, LOGAUTO, is described, which when installed, makes it possible for users to instantly execute any one of the five procedures by simply typing its name.

---

*Research Technician and **Chemist, Metallurgical Chemistry Section, Mineral Sciences Laboratories, CANMET, Energy, Mines and Resources Canada, 555 Booth Street, Ottawa, Ontario, K1A 0G1.

CINQ PROCÉDURES CCL QUI SIMPLIFIENT GRANDEMENT LE TRANSFERT DES FICHIERS ENTRE DES MICRO-ORDINATEURS ET LE CYBER

par

P.L. Bédard* et K. Bartels**

RÉSUMÉ

Les auteurs de ce rapport décrivent cinq routines du langage de contrôle de CYBER (CCL), qui simplifient considérablement le transfert des fichiers entre des micro-ordinateurs et l'unité centrale CYBER, en exemptant l'utilisateur de l'obligation d'introduire une série de chaînes de commande compliquées à chaque transfert.

Les procédures ULXFNC, ULXFC, ULNXFC, DLXF et DLNXF, qui produisent des messages guide-opérateur et proposent des choix de réponses, permettent donc aux utilisateurs de trouver aisément les paramètres corrects que réclament le logiciel de transmission de données XMODEM et le logiciel de conversion de fichiers XFORMAT. Les auteurs décrivent chaque procédure en détails et donnent toutes les instructions nécessaires à la mise en place des programmes, de sorte qu'il est facile ensuite, pour les utilisateurs, d'appeler ces procédures chaque fois qu'ils entrent en communication avec CYBER. Il est aussi question d'une sixième procédure, LOGAUTO, qui, une fois installée, permettra aux utilisateurs d'exécuter instantanément n'importe laquelle des cinq premières, simplement en introduisant son nom au clavier.

---

* Technicien en recherche et **Chimiste, Section de la chimie métallurgique, Laboratoires des sciences minérales, CANMET, ministère de l'Énergie, des Mines et des Ressources, 555, rue Booth, Ottawa (Ont.), K1A 0G1.

INTRODUCTION

In recent years, there have been a large number of microcomputer acquisitions within E.M.R. As in our case, many of these microcomputers are used to communicate, and more importantly, to effect file transfers to and from the CYBER. Transfers to the CYBER are usually made in order to process data on the mainframe using such routines as EMRLIB or DISPPLA or high level languages such as FORTRAN or COBOL. In other cases, the CYBER may simply be used as a mass storage device for backup purposes or for intermittent storage from one to another microcomputer user when their direct connection by modem is not practical.

Typically, a microcomputer may be used within a laboratory for data acquisition, thereby building a data file which is to be used as input to a mainframe FORTRAN program. Therefore, it is necessary to transfer such a file to the CYBER so that it may be processed further in the high-level language. Conversely, a high-level language CYBER routine may write a file of computed results to mass storage and it may then be necessary to return this file to the laboratory/office for further use, such as inclusion in a report under a word processing format on the microprocessor.

In the first case, the file has to be transferred from diskette via the microcomputer to mass storage on the CYBER, and in the second case, the file has to be transferred from mass storage on the CYBER to diskette storage via the microcomputer.

The examples given in this report are those for a microcomputer using a MODEM-7 communications program under the control of the CP/M operating system. However, the PROC's (PROCedures) in this report would be equally suitable for use with other microcomputers such as an APPLE or compatible running under DOS 3.3 using APPLE-COM, or an IBM-PC or compatible running under PC-DOS using PC-TALK or CROSSTALK. Details for these, may be found in (1) or other CSC user guides.

The examples of XFORMAT in this report specify conversions between CYBER DISPLAY and XMODEM (Microcomputer ASCII) format. However, the PROC's may be re-written by individual users to other defaults, such as conversions from XMODEM to EBCDIC code. Only a limited number of the many parameters available for each of these programs have been defined within each XMODEM or XFORMAT command string. We have found these sufficient for all our purposes. However, even with only a few parameters to define,

each file transfer requires keyboard entry of a number of tedious and very 'user unfriendly' command strings. For example, to send a file called 'myfile', formatted in microcomputer ASCII, to the CYBER, convert it to DISPLAY code and catalog it to mass storage, requires the following five command strings:

```
BEGIN,XMODEM,,myfile,R,PROT=MODEM,Q
ATTACH,XF,BINXFORMAT,MR=1
XF,I=myfile(XMODEM),O=lfn(DISPLAY),LF=0
CATALOG, LFN,PFN,ID=IDD,RP=RPP,CY=CYY
RETURN,XF
```

These five command string have to be repeated each time such a file transfer is required. Explanation of the above command strings may be found in (1) and (2).

Over the last two years we have done many file transfers to and from the CYBER and have identified five particular usages which have covered all of our file transfer requirements. Consequently, we have written five CCL (Cyber Control Language) PROC's and an automatic logon procedure that make file transfers much easier. The PROC's query the user for the required parameters and then load these into the command strings before processing them. A limited level of help is provided within each procedure when users are prompted for the variables. We feel that other users of microcomputers might benefit from these procedures as they provide a 'user-friendly' interface and avoid the necessity of typing in long command strings such as those given above.

CYBER COMMUNICATION SOFTWARE

The CYBER communications software with which most microcomputer users are concerned, i.e., which enables file transfers to and from the micro to the mainframe, are XMODEM and XFORMAT. The following short summaries have been taken from the "CYBER / MICROCOMPUTER FILE TRANSFER USER'S GUIDE" (1):

a) XMODEM - is a CYBER program which transfers a file with error-checking, to and from a microcomputer over asynchronous communications lines. The file transfer protocol was originally developed by Ward Christensen who also wrote the first of the family of popular MODEM communications programs which implemented tne protcol on the CP/M operating system.

A file is transferred as a sequence of 128-byte blocks. Each block has header and trailer information to allow error detection and retransmission of a bad block before the next block is sent.

b) XFORMAT - is a CYBER program which transforms CYBER files which have been stored in various formats into a standard format for transmission to a microcomputer by the XMODEM utility. It also transforms RECEIVED XMODEM microcomputer files into different storage formats for use on the CYBER.

XFORMAT converts files on the CYBER between different characters sets (DISPLAY, ASCII, EBCDIC) and different physical storage formats (6-bit, 7-bit, or 8-bit characters stored in many different ways within CYBER words or cros_ing word boundaries). It also transforms embedded carriage control characters in microcomputer print files to the column 1 carriage control that is required on the CYBER, expands tab settings with blank spacing, and processes backspace characters. Input characters are processed in units of lines. They are accumulated and partially translated until a line delimiter is found at which time a completely translated output is generated.

GENERAL STRUCTURE OF THE MICROCOMPUTER/CYBER CCL FILE TRANSFER PROCEDURES

The five file transfer PROC's: ULXFNC, ULXFC, ULNXFC, DLXF and DLNXF have several common features. When they are invoked, a short paragraph describing their use, is written to the terminal. Except for ULXFNC, a further help level may be chosen explaining the parameters required by each PROC.

Once the required parameters have been selected, the user has an opportunity to review the selections and to re-enter them in the event of an error. When the user has indicated that the proper selections have

been made by answering 'YES' to the re-viewing prompt, the PROC will begin executing. During execution, the PROC's issue a number of informative statements back to the terminal as they step through the several command strings, thereby informing the user as to the status of the execution. In the case where an ATTACHMENT and a READ from a CYBER mass storage file has taken place, the user has the option to PURGE that file.

NAMING OF THE INDIVIDUAL PROC'S

a) Procedures for file transfer from the MICROCOMPUTER to the CYBER

These procedures have been named according to the direction of their transfer. The direction from the MICROCOMPUTER to the CYBER is 'Up' and the 'UL' part of the PROC name refers to 'Up-Loading'.

b) Procedures for file transfer from the CYBER to the MICROCOMPUTER

In this transfer, CYBER to MICROCOMPUTER, the direction has been defined as 'Down', and the 'DL' part of the name refers to 'Down-Loading'.

c) Invoking of XFORMAT

Except in the case where the CYBER is used only as a mass storage device, XFORMAT must be used to convert the microcomputer ASCII file to DISPLAY code, when Up-Loading, or to convert the DISPLAY code to microcomputer ASCII when Down-Loading. Thus XF refers to invoking XFORMAT and NXF refers to not-invoking XFORMAT.

d) Cataloging a file that has been Up-Loaded

We have assumed that an Up-Load to the CYBER always requires cataloging so that the file is available to the user for more than one day. Cataloging is therefore a default for the 'UL' type PROC's. Thus the ID (identifier), CY (cycle number) and RP (retention period) may be defined by the user.

The naming conventions have been summarized in Table 1. This summary also prints out to the screen if the user takes the option of installing the procedure LOGAUTO (described in a later Section) as an automatic logon subroutine (2).

Table 1 - Summary of PROC naming conventions related to PROC usage

Uploading ('UL'): Transferring a File From the MICRO to the CYBER
1) [ULXFNC] - XFORMAT - NO CATALOG - BATCH TO INPUT QUEUE 2) [ULXFC] - XFORMAT - CATALOG FILE (PFN,ID,CY,RP) 3) [ULNXFC] - NO XFORMAT CATALOG FILE (PFN,ID,CY,RP)
Downloading ('DL'): Transferring a File From the CYBER to the MICRO
4) [DLXF] - XFORMAT - ATTACHES FILE (PFN,ID,CY) 5) [DLNXF] - NO XFORMAT ATTACHES FILE (PFN,ID,CY)

Comments for Individual Procedures

a) ULXFNC - (see Appendix I)

This PROC should be used when a file is to be transferred from the MICROPROCESSOR to the CYBER and immediately submitted to the INPUT queue for processing. It automatically invokes XFORMAT to convert the microprocessor ASCII file into a CYBER DISPLAY file prior to batching it to the INPUT queue. In addition, it automatically purges the 'lfn' (logical file name) which was involved in the transfer under the XMODEM procedure. It does not catalog a file.

Caution: During our use of XFORMAT when Up-Loading, we discovered that XFORMAT puts a blank line at the very end of the CYBER DISPLAY file. This of course, wreaks havoc when trying to execute high-level language source or data code because the file now does not contain a proper 'END OF RECORD' (*EOR) or 'END OF FILE' (*EOF) in the expected location (i.e., All CYBER files are terminated with *EOR/*EOF). Therefore, we have added additional code to ULXFNC to automatically edit the converted file using the MSUEDIT command, thereby deleting the blank line from the end of the file prior to submitting the file to the INPUT queue.

b) ULXFC - (see Appendix II)

This PROC should be used when a file is to be transferred from the MICROPROCESSOR to the CYBER and cataloged, prior to further viewing or processing on the mainframe. It automatically invokes XFORMAT to convert the microprocessor ASCII file into a CYBER DISPLAY file. In addition, it automatically purges the 'lfn' which was involved in the transfer under the

XMODEM procedure and catalogs the DISPLAY file with the following user-specified parameters:

PFN, ID, CY and RP
(permanent filename, identifier, cycle number and retention period)

Caution: The same caution as that with ULXFNC applies to this procedure also. The same code has been added as mentioned above in order to delete the last blank line.

c) ULNXFC - (see Appendix III)

This PROC should be used when a file is to be transferred from the MICROPROCESSOR to the CYBER and simply cataloged, i.e., "no XFORMAT" is invoked. In this case, the CYBER acts simply as a mass storage device for the XMODEM file (e.g., as backup for microprocessor files on the CYBER mass storage facilities or as an intermediate storage device for two or more microcomputer users who may have communication facilities to the CYBER, but not to each other). Therefore, a conversion to CYBER DISPLAY code is unnecessary as no mainframe processing or viewing of the file is intended. In addition, it automatically reNAMES and reCATALOGs the XMODEM file with the following user-specified parameters:

PFN, ID, CY and RP
(permanent filename, identifier, cycle number and retention period)

In this way, XMODEM default values are changed to those specified by the user.

d) DLXF - (see Appendix IV)

This PROC should be used when a file is to be transferred from the CYBER to the MICROPROCESSOR and the file resident on the CYBER is in a format other than microprocessor ASCII. As before, the PROC prompts will ask for a PFN, ID and CYcle number. The file to be transferred can have three dispositions:

1. The file is already ATTACHED to the terminal (it has a 'LOCAL' disposition).

2. The file is not yet ATTACHED (it is still a permanent mass storage file). In this case, there are two options when accessing a particular cycle of the required file:

   i) The cycle with the highest cycle number is attached when 'LAST' is entered to the 'CY' prompt.
   ii) The user may specify a specific cycle number.

After the file has been Down-Loaded, the user is then given the opportunity of RETAINing or PURGEing the file.
Caution: During our use of XFORMAT when Down-Loading, we discovered that XFORMAT deletes all 'END OF RECORDS' (*EOR's) and 'END OF FILES' (*EOF's) when creating an XMODEM file. Therefore, we have added additional code to DLXF to automatically edit the DISPLAY file using the MSUEDIT command, thereby changing all lines beginning with '*EOR' to 'STAREOR' and '*EOF' to 'STAREOF' before invoking XFORMAT.

e) DLNXF - (see Appendix V)

This PROC should be used when a file, stored on the CYBER, is already in microprocessor ASCII format; e.g. one that was Up-Loaded by the PROC ULNXFC (see (c) above). As before, the PROC prompts will ask for a PFN, ID, and CY. The file to be transferred can have three dispositions:

1. The file is already ATTACHED to the terminal (it has a 'LOCAL' disposition).

2. The file is not yet ATTACHED (it is still a permanent mass storage file). In this case, there are two options when accessing a particular cycle of the required file:

   i) The cycle with the highest cycle number is attached when 'LAST' is entered to the 'CY' prompt.
   ii) The user may specify a specific cycle number.

After the file has been Down-Loaded, the user is then given the opportunity of RETAINing or PURGEing the file.

CREATION AND INSTALLATION OF CCL LIBRARY 'MYLIB'

Users may create and install the necessary CCL library 'MYLIB', by following this example adapted from the 'CCL USER'S GUIDE' (3):

```
COMMAND- attach,in,myprocs,id=myid                                    (i)
 PF CYCLE NO. = 001
COMMAND- request,mylib.*pf                                            (ii)
COMMAND- connect,input
COMMAND- editlib                                                      (iii)
library(mylib,new)                                                    (iv)
add(*,in,al=1)                                                        (v)
finish.
endrun.
%eof
COMMAND- catalog,mylib,id=myid,rp=999                                 (vi)
   FILE SIZE 00000012 PRUS
 INITIAL CATALOG
CT ID=      MYID PFN=MYLIB
CT CY= 001  00000768 WORDS.SN=PUBLIC :
COMMAND-
```

i) Attach the permanent file 'MYPROC' containing the five PROC's

ii) Request permanent file space for 'MYLIB' library file.

iii) Call EDITLIB to create a user library file.

iv) The library directive - LIBRARY,NEW - statement indicates that the file 'MYLIB' is a new library (i.e., a creation run).

v) The - ADD - directive with an asterisk (*) means: add all the procedures from file INPUT to the library; the access level (AL) is set to one, permitting the procedures to be called without the use of the BEGIN statement. (Refer to Section 4-40 in the NOS/BE Reference Manual for details on the use of EDITLIB) (3).

vi) Catalog the file with a permanent file name of 'MYLIB', file ID, 'MYID' and infinite retention.

Note: The five PROC's must be resident on a CYBER file called 'MYPROCS', and be available to the user at the time of installing 'MYLIB' as per the ATTACH parameters in the first line of the example. Users should contact the authors in order to get a mass storage copy of the file containing the five PROC's as listed in Appendix I to V.

CREATION AND IMPLEMENTATION OF 'LOGAUTO'

In order to automatically load the five PROC's each time a user logs on, it is suggested that the additional PROC 'LOGAUTO' be implemented. Thus, when a user logs onto the CYBER, the 'LOGAUTO' procedure will automatically attach and implement the CCL library MYLIB (see previous section on installation of MYLIB), containing the five PROC's discussed above. By using 'LOGAUTO', the five PROC's are made instantly available to the user for execution simply by typing the PROC name. As part of 'LOGAUTO', an optional help menu, in which the available PROC's, their usage and command form is summarized, can be printed at the terminal.

To install the 'LOGAUTO' feature described above, users must first create a 'LOGAUTO' file such as given in Appendix VI, edit it according to their own preferences (e.g. LOGON message) using TXTEDIT or a word processor, and store it as permanent file on the CYBER under their own ACCOUNT number using a filename (e.g., 'MYLOGAUTO') and an ID (e.g., 'MYID'). The user then proceeds to LOGOUT. The next logon has to be without prompting following the LOGIN message according to the following:

```
login,username,password,sup,params
ENTER KEYWORD=VALUE OR END
?pf=mylogauto
ENTER KEYWORD=VALUE OR END
?id=myid
ENTER KEYWORD=VALUE OR END
?end
(user entries are indicated in lower case)
```

The above LOGIN command instructs the system to attach a LOGIN procedure file with PEN of 'LOGAUTO' and an ID of 'MYID' and execute the commands contained in that file (2). Hereafter, 'LOGAUTO' will automatically execute each time the user logs on to the CYBER.
Note: The file 'LOGAUTO', as given in Appendix VI, is also available from the CSC.

ACKNOWLEDGEMENTS

We wish to thank the following for some valuable help and suggestions: R.F. Pilgrim (CANMET) for help with CCL and EDITLIB syntax, Mike Wenzowski (CSC) for INTERCOM and interactive CCL useage tips and I. Cottrell, D. Carriere (CSC) and C.A. Hamer (CANMET) for testing the PROC's during development.

REFERENCES

1. "CYBER / MICROCOMPUTER FILE TRANSFER USER'S GUIDE", Published by Computer Science Centre, Energy, Mines and Resources Canada, Ottawa, January 1984, Revision A.

2. "TIMESHARING USER'S GUIDE", Published by Computer Science Centre, Energy, Mines and Resources Canada, Ottawa, November 1983.

3. "CCL USER'S GUIDE", Published by Computer Science Centre, Energy, Mines and Resources Canada, Ottawa, November 1983.

APPENDIX I

ULXFNC: CCL procedure for file transfer from microcomputer to CYBER with XFORMAT and automatic batch to INPUT queue.

```
PROC,ULXFNC.
.*********************************************************
.*                                                       *
.*         FILE TRANSFER FROM MICRO TO CYBER             *
.*   [XFORMAT - NO CATALOG - BATCH TO INPUT QUEUE]       *
.*                                                       *
.* VERSION INFORMATION:                                  *
.* ORIGINAL  V 1.0  BY P. L. BEDARD   EMR   JAN 1985     *
.*********************************************************
REMARK. ================== [GENERAL INFO.] ==================
REMARK. [ULXFNC]-> PROCEDURE "ULXFNC" SENDS A FILE FROM A
REMARK.            MICRO TO THE CYBER, EXECUTES XFORMAT
REMARK.            AND BATCHES THE FILE TO THE INPUT QUEUE
REMARK.
REMARK. ================== [CCL PROCEDURE "ULXFNC"] ==================
REMARK.          [XFORMAT - NO CATALOG - BATCH TO INPUT QUEUE]
REMARK.
REMARK. [ULXFNC]-> DOING XMODEM  ... PLEASE WAIT ...
BEGIN,XMODEM,,XNAME,R,T=500.
REMARK. [ULXFNC]-> DOING XFORMAT ... PLEASE WAIT ...
ATTACH,XF,BINXFORMAT,MR=1.
XF,I=XNAME(XMODEM),O=B(DISPLAY),LF=0.
REQUEST,C,*Q.
MSUEDIT,I=DIR.
PURGE,XNAME.
RETURN,XF,XNAME,PROCFIL,DAYFL,DAYFIL,OUTPUT.
REMARK. [ULXFNC]-> BATCHING FILE ... PLEASE WAIT ...
BATCH,C,INPUT.
REMARK. [ULXFNC]-> END OF PROCESSING !
REVERT.
.DATA,DIR.
LENGTH 140.
EDIT,B,FR 100.
D,*L.
/STAR00/=/*EOR00/,100-*L,A.
S,C,NS.
.*************** END OF CCL PROC "ULXFNC" ***************
```

APPENDIX II

ULXFC: CCL procedure for file transfer from microcomputer to CYBER with XFORMAT and cataloging on CYBER.

```
.PROC,ULXFC.
.*********************************************************
.*                                                       *
.*         FILE TRANSFER FROM MICRO TO CYBER             *
.*       [XFORMAT - CATALOG ON FILE SUCH THAT]           *
.*            "PFN,ID=FID,RP=RPP,CY=CYY"                 *
.*                                                       *
.* VERSION INFORMATION:                                  *
.* ORIGINAL  V 1.0  BY P. L. BEDARD   EMR   JAN 1985 *
.*********************************************************
REMARK. ================== [GENERAL INFO.] ==================
REMARK. [ULXFC ]-> PROCEDURE "ULXFC" PROVIDES INTERACTIVE
REMARK.            INPUT FOR USE WITH XMODEM AND XFORMAT
REMARK.            WHEN SENDING A FILE FROM A MICRO TO THE
REMARK.            CYBER AND CATALOGING IT.
REMARK.
SET,R1G=0.
WHILE,R1G.EQ.0,LOOP.
PROC?
ENDW,LOOP.
RETURN,PROC.
REVERT.
.DATA,PROC
.PROC,PROC*I,
PFN"(? FOR HELP)"=(*A),
FID"(? FOR HELP)"=(*A),
CCY"(? FOR HELP)"=(*A),
RPP"(? FOR HELP)"=(*A).
.*********************************************************
.HELP,PFN
========================== [PFN HELP] ==========================
[ULXFC ]-> FILE NAME ENTERED HERE MAY BE THE SAME AS THAT ON
           THE MICRO. IF YOUR SYSTEM REQUIRES THAT THE FILE
           NAME INCLUDE A PERIOD, THE NAME ENTERED MUST BE
           APPENDED WITH A $ (DOLLAR) SIGN.
===
.HELP,FID
========================== [FID HELP] ==========================
[ULXFC ]-> THIS PROCEDURE ALLOWS A FILE ID TO BE SPECIFIED
           PRIOR TO CATALOGING THE FILE ON THE CYBER.
===
.HELP,CCY
========================== [CYY HELP] ==========================
[ULXFC ]-> THIS PROCEDURE ALLOWS A CYCLE NUMBER TO BE SPECIFIED
           PRIOR TO CATALOGING THE FILE ON THE CYBER.
===
.HELP,RPP
========================== [RPP HELP] ==========================
[ULXFC ]-> THIS PROCEDURE ALLOW A FILE RETENTION PERIOD TO BE
           SPECIFIED PRIOR TO CATALOGGING THE FILE ON CYBER.
```

```
==
.ENDHELP
REMARK.
REMARK. ##PFN      = PFN
REMARK. ##FID      = FID
REMARK. ##CCY      = CCY
REMARK. ##RPP      = RPP
SET R1G=1.
IFNO,SKP. [ULXFC ]-> ARE THESE ACCEPTABLE?
SET,R1G=0.
REMARK.
REVERT.
ENDIF,SKP.
REMARK. ================== [CCL PROCEDURE "ULXFC "] ==================
REMARK. [XFORMAT  - CATALOG ("PFN,ID=FID,CY=CCY,RP=RPP") - NO BATCH]
RETURN,C,LFN.
REMARK. [ULXFC ]-> DOING XMODEM  ... PLEASE WAIT ...
BEGIN,XMODEM,,B,R,T=500.
REMARK. [ULXFC ]-> DOING XFORMAT ... PLEASE WAIT ...
ATTACH,XFORMAT,BINXFORMAT,MR=1.
XFORMAT,I=B(XMODEM),O=C(DISPLAY),LF=0.
REQUEST,C,*Q.
REMARK. [ULXFC ]-> DOING MSUEDIT ... PLEASE WAIT ...
REMARK. [ULXFC ]-> LENGTH HAS BEEN SET TO 140 COLUMNS ...
REQUEST,LFN,*PF.
MSUEDIT,I=DIR.
CATALOG,LFN,PFN,ID=FID,RP=RPP,CY=CCY.
PURGE,B.
RETURN,B,C.
REMARK. [ULXFC ]-> END OF PROCESSING !
REVERT.
.DATA,DIR.
LENGTH 140.
EDIT,C,FR,100.
/STAR00/=/*EOR00/,100-*L,A.
D,*L.
S,LFN,NS.
.*************** END OF CCL PROC "ULXFC"  ***************
```

APPENDIX III

ULNXFC: CCL procedure for file transfer from microcomputer without XFORMAT and cataloging on the CYBER.

```
.PROC,ULNXFC.
.*********************************************************
.*                                                       *
.*         FILE TRANSFER FROM MICRO TO CYBER             *
.* [NO XFORMAT - CATALOGUE FILE ON CYBER SUCH THAT]      *
.*            "PFN,ID=FID,RP=RPP,CY=CYY"                 *
.*                                                       *
.* VERSION INFORMATION:                                  *
.* ORIGINAL  V 1.0  BY P. L. BEDARD   EMR   JAN 1985     *
.*********************************************************
REMARK. ================== [GENERAL INFO.] ==================
REMARK. [ULNXFC]-> PROCEDURE "ULNXFC" PROVIDES INTERACTIVE
REMARK.            INPUT FOR USE WITH XMODEM WHEN SENDING
REMARK.            A FILE FROM A MICRO TO THE CYBER AND
REMARK.            CATALOGING IT.
REMARK.
SET,R1G=0.
WHILE,R1G.EQ.0,LOOP.
PROC?
ENDW,LOOP.
RETURN,PROC.
.DATA,PROC
.PROC,PROC*I,
PFN"(? FOR HELP)"=(*A),
FID"(? FOR HELP)"=(*A),
CCY"(? FOR HELP)"=(*A),
RPP"(? FOR HELP)"=(*A).
.HELP,PFN
========================== [PFN HELP] ==========================
[ULNXFC]-> FILE NAME ENTERED HERE MAY BE THE SAME AS THAT ON
           THE MICRO. IF YOUR SYSTEM REQUIRES THAT THE FILE
           NAME INCLUDE A PERIOD, THE NAME ENTERED MUST BE
           APPENDED WITH A $ (DOLLAR) SIGN.
===
.HELP,FID
========================== [FID HELP] ==========================
[ULNXFC]-> THIS PROCEDURE ALLOWS A FILE ID TO BE SPECIFIED
           PRIOR TO CATALOGING THE FILE ON THE CYBER.
===
.HELP,CCY
========================== [CYY HELP] ==========================
[ULNXFC]-> THIS PROCEDURE ALLOWS A CYCLE NUMBER TO BE SPECIFIED
           PRIOR TO CATALOGING THE FILE ON THE CYBER.
===
.HELP,RPP
========================== [RPP HELP] ==========================
[ULNXFC]-> THIS PROCEDURE ALLOW A FILE TO BE SPECIFIED PRIOR
           TO CATALOGGING THE FILE ON CYBER.
===
.ENDHELP
```

```
REMARK.
REMARK. ##PFN      = PFN
REMARK. ##FID      = FID
REMARK. ##CCY      = CCY
REMARK. ##RPP      = RPP
SET R1G=1.
IFNO,SKP. [ULNXFC]-> ARE THESE ACCEPTABLE?
SET,R1G=0.
REMARK.
REVERT.
ENDIF,SKP.
REMARK. ================== [CCL PROCEDURE "ULNXFC"] ==================
REMARK.        [CATALOG - SIMPLE FILE TRANSFER FROM MICRO TO CYBER]
REMARK.            [CATALOGED AS "PFN,ID=FID,CY=CCY,RP=RPP.]
REMARK.
REMARK. [ULNXFC]-> DOING XMODEM  ... PLEASE WAIT ...
BEGIN,XMODEM,,PFN,R,T=500.
RENAME,PFN,ID=FID,CY=CCY,RP=RPP.
RETAIN,MYLIB,PFN.
FILES.
REMARK. [ULNXFC]-> END OF PROCESSING !
EXIT.
.*************** END OF CCL PROC "ULNXFC" ***************
```

APPENDIX IV

DLXF: CCL procedure for file transfer from CYBER to microcomputer with XFORMAT.

```
.PROC,DLXF.
.*********************************************************
.*                                                       *
.*         FILE TRANSFER FROM CYBER TO MICRO             *
.*   [XFORMAT - REQUIRES FILE PARAMETERS SUCH THAT]      *
.*                  "PFN,ID=FID,CY=CYY"                  *
.*                                                       *
.* VERSION INFORMATION:                                  *
.* ORIGINAL  V 1.0  BY P. L. BEDARD   EMR   JAN 1985     *
.*********************************************************
REMARK. ================== [GENERAL INFO.] ==================
REMARK. [ DLXF ]-> PROCEDURE "DLXF" PROVIDES INTERACTIVE
REMARK.            INPUT FOR USE WITH XMODEM AND XFORMAT
REMARK.            WHEN SENDING A FILE FROM THE CYBER TO
REMARK.            A MICRO.
REMARK.
SET,R1G=0.
WHILE,R1G.EQ.0,LOOP.
PROC?
ENDW,LOOP.
RETURN,PROC.
REVERT.
.DATA,PROC
.PROC,PROC*I,
PFN"(? FOR HELP)"=(*A),
FID"(? FOR HELP)"=(*A),
CCY"(? FOR HELP)"=(*A).
.*********************************************************
.HELP,PFN
========================= [PFN HELP] =========================
[ DLXF ]-> FILE NAME ENTERED HERE MAY BE THE SAME AS THAT ON
           THE MICRO. IF YOUR SYSTEM REQUIRES THAT THE FILE
           NAME INCLUDE A PERIOD, THE NAME ENTERED MUST BE
           APPENDED WITH A $ (DOLLAR) SIGN.
==
.HELP,FID
========================= [FID HELP] =========================
[ DLXF ]-> THIS PROCEDURE ALLOWS A FILE ID TO BE SPECIFIED
           PRIOR TO CATALOGING THE FILE ON THE CYBER.
==
.HELP,CCY
========================= [CYY HELP] =========================
[ DLXF ]-> THIS PROCEDURE ALLOWS A CYCLE NUMBER TO BE SPECIFIED
           PRIOR TO CATALOGING THE FILE ON THE CYBER.
==
.ENDHELP
REMARK.
REMARK. ##PFN     = PFN
REMARK. ##FID     = FID
REMARK. ##CCY     = CCY
```

```
SET R1G=1.
IFNO,SKP. [ DLXF ]-> ARE THESE ACCEPTABLE?
SET,R1G=0.
REMARK.
REVERT.
ENDIF,SKP.
REMARK. ================== [CCL PROCEDURE * DLXF *] ==================
REMARK.          [XFORMAT - FILE TRANSFER FROM CYBER TO MICRO]
REMARK.              [ALL "EOR" WILL BE CHANGED TO "STAR"]
REMARK.
RETURN,PFN.
IFE FILE(ADL,AS),ATTPF.
REWIND,ADL.
ELSE,ATTPF.
IFE,$CCY$=$LAST$,IKJ.
ATTACH,ADL,PFN,ID=FID.
ELSE,IKJ.
ATTACH,ADL,PFN,ID=FID,CY=CCY.
ENDIF,IKJ.
ENDIF,ATTPF.
MSUEDIT,I=DIR.
REMARK. [ DLXF ]-> DOING XFORMAT ... PLEASE WAIT ...
ATTACH,XF,BINXFORMAT,MR=1.
XF,I=A(DISPLAY),O=B(XMODEM),LF=1.
REMARK. [ DLXF ]-> DOING XMODEM  ... PLEASE WAIT ...
RETURN,XF.
BEGIN,XMODEM,,B,S,T=500.
REMARK.
IFNO(L2) DO YOU WANT TO 'SAVE' "PFN"?
REMARK.
REMARK. [ DLXF ]-> FILE "PFN" IS BEING PURGED.
PURGE,ADL.
ENDIF,L2.
RETAIN,MYLIB.
REMARK. [ DLXF ]-> END OF PROCESSING !
EXIT.
.DATA,DIR.
LENGTH 140.
EDIT,ADL,FR 100.
/EOR00/=/STAR00/,100-*L,A.
S,A,NS
.*************** END OF CCL PROC "DLXF"   ***************
```

APPENDIX V

DLNXF: CCL procedure for file transfer from CYBER to microcomputer without XFORMAT.

```
.PROC,DLNXF.
.*********************************************************
.*                                                       *
.*         FILE TRANSFER FROM CYBER TO MICRO             *
.*   [NO XFORMAT - REQUIRES PRIOR LOGON TO CYBER]        *
.*                  "PFN,ID=FID,CY=CYY"                  *
.*                                                       *
.* VERSION INFORMATION:                                  *
.* ORIGINAL  V 1.0  BY P. L. BEDARD   EMR   JAN 1985 *
.*********************************************************
REMARK. ================== [GENERAL INFO.] ==================
REMARK. [DLNXF ]-> PROCEDURE "DLNXF" PROVIDES INTERACTIVE
REMARK.            INPUT FOR USE WITH XMODEM WHEN SENDING
REMARK.            A FILE FROM THE CYBER TO THE MICRO.
REMARK.
SET,R1G=0.
WHILE,R1G.EQ.0,LOOP.
PROC?
ENDW,LOOP.
RETURN,PROC.
REVERT.
.DATA,PROC
.PROC,PROC*I,
PFN"(? FOR HELP)"=(*A),
FID"(? FOR HELP)"=(*A),
CCY"(? FOR HELP)"=(*A).
.*********************************************************
.HELP,PFN
========================= [PFN HELP] =========================
[DLNXF ]-> FILE NAME ENTERED HERE MAY BE THE SAME AS THAT ON
           THE MICRO. IF YOUR SYSTEM REQUIRES THAT THE FILE
           NAME INCLUDE A PERIOD, THE NAME ENTERED MUST BE
           APPENDED WITH A $ (DOLLAR) SIGN.
===
.HELP,FID
========================= [FID HELP] =========================
[DLNXF ]-> THIS PROCEDURE ALLOWS A FILE ID TO BE SPECIFIED
           PRIOR TO CATALOGING THE FILE ON THE CYBER.
===
.HELP,CCY
========================= [CYY HELP] =========================
[DLNXF ]-> THIS PROCEDURE ALLOWS A CYCLE NUMBER TO BE SPECIFIED
           PRIOR TO CATALOGING THE FILE ON THE CYBER.
===
.ENDHELP
REMARK.
REMARK. ##PFN     = PFN
REMARK. ##FID     = FID
REMARK. ##CCY     = CCY
SET R1G=1.
```

```
IFNO,SKP. [DLNXF ]-> ARE THESE ACCEPTABLE?
SET,R1G=0.
REMARK.
REVERT.
ENDIF,SKP.
REMARK.
REMARK. ================ [CCL PROCEDURE "DLNXF "] ================
REMARK.    [NO XFORMAT - SIMPLE FILE TRANSFER FROM CYBER TO MICRO]
REMARK.
IFE FILE(PFN,PF),ATTPF.
REWIND,PFN.
ELSE,ATTPF.
IFE,$CCY$=$LAST$,IJB.
ATTACH,PFN,ID=FID.
ELSE,IJB.
ATTACH,PFN,ID=FID,CY=CCY.
ENDIF,IJB.
ENDIF,ATTPF.
REMARK. [DLNXF ]-> DOING XMODEM  ... PLEASE WAIT ...
BEGIN,XMODEM,,PFN,S,T=500.
IFNO(L2) DO YOU WANT TO 'SAVE' "PFN"?
REMARK.
REMARK. [DLNXF ]-> FILE "PFN" IS BEING PURGED.
PURGE,PFN.
ENDIF,L2.
RETAIN,MYLIB.
FILES.
REMARK. [DLNXF ]-> END OF PROCESSING !
EXIT.
.*************** END OF CCL PROC "DLNXF"  ***************
```

APPENDIX VI

LOGAUTO: Procedure for automatic attachment and implementation of CCL communication PROC's upon logging on to CYBER.

```
.PROC,LOGAUTO.
.*****************************************************
.*                                                   *
.*  AUTOMATIC LOGON PROCEDURE AND CCL 'PROC' MENUS   *
.*          FOR MICRO/CYBER FILE TRANSFERS           *
.*                                                   *
.*****************************************************
DAYFILE,OFF.
REMARK.
REMARK. )) WELCOME TO INTERACTIVE MICRO/CYBER FILE TRANSFER PROCEDURES ((
REMARK.
REMARK. [LOGAUTO]-) AVAILABLE PROCEDURES
REMARK.              ULXFNC   ULXFC   ULNXFC   DLXF   DLNXF
IFNO(ANS) [LOGAUTO]-) TYPE 'YES' TO 'SKIP' HELP:
REMARK.
REMARK. [LOGAUTO]-) AVAILABLE CCL MICRO/CYBER FILE TRANSFER PROCEDURES
REMARK.
REMARK. UPLOADING ('UL')  : TRANSFERING A FILE FROM THE MICRO TO THE CYBER
REMARK.    1) [ULXFNC] -    XFORMAT - NO CATALOG - BATCH TO INPUT QUEUE
REMARK.    2) [ULXFC ] -    XFORMAT -    CATALOG FILE  (PFN,ID,CY,RP)
REMARK.    3) [UPNXFC] - NO XFORMAT -    CATALOG FILE  (PFN,ID,CY,RP)
REMARK.
REMARK. DOWNLOADING ('DL'): TRANSFERING A FILE FROM THE CYBER TO THE MICRO
REMARK.    4) [DLXF  ] -    XFORMAT -    ATTACHES FILE (PFN,ID,CY)
REMARK.    5) [DLNXF ] - NO XFORMAT -    ATTACHES FILE (PFN,ID,CY)
REMARK.
REMARK.    WHERE: UL     = A FILE TRANSFER FROM A MICRO TO THE CYBER
REMARK.           DL     = A FILE TRANSFER FROM THE CYBER TO A MICRO
REMARK.           XF/NXF = XFORMAT/NO XFORMAT
REMARK.           C/NC   = CATALOG/NO CATALOG
REMARK.
ENDIF(ANS)
RETURN,MYLIB.
ATTACH,MYLIB,ID=LIB,MR=1.
LIBRARY,MYLIB.
RETURN,CONTROL.
DAYFILE,ON.
REVERT.
```